intelligence module could propagate to the clones changes and additions to its datastore as they occur or, in other embodiments, at periodic intervals.

Although this example has described modules running on computers located near certain network areas, it is specifically noted that in certain embodiments the computers may be located elsewhere. For example, with growth of the system it might be desirable to add additional modules to expand the capability of the headquarters. In such a case the additional modules may run on additional general purpose computers that are located at or near DDS headquarters rather than at or near the network areas.

Hardware and Software

The software, modules, objects, components and other code and/or software elements described above could be written, for example, using an object-oriented language known in that art such as Objective-C, Java, or C#. Phrases such as "component," "module," and "object," as used herein, may refer, for example, to program code written as a class using an object-orientated programming language and instantiated into an object using techniques known in the art of object-orientated programming.

The modules or other software could run, for example, on computers including personal computers or workstations such as Power Macintosh G4s or Dell Dimensions running operating systems such as Apple OS X, Microsoft Windows XP, or Linux, perhaps further including support for Java. The modules or other software could also run, for example, on PDAs, cellular telephones, DVB-T receivers, or the like running an operating system such as Microsoft Windows CE or Symbian EPOC, perhaps with support for Java. Speaking, more generally, the modules or other software could run on a general purpose computer.

The phrases "general purpose computer," "computer," and the like, as used herein, refer but are not limited to an engineering workstation, PC, Macintosh, PDA, web-enabled cellular phone and the like running an operating system such as OS X, Linux, Windows CE, Windows XP, Symbian EPOC, or the like. The phrases "general purpose computer," "computer," and the like also refer, but are not limited to, one or more processors operatively connected to one or more memory or storage units, wherein the memory or storage may contain data, algorithms, and/or program code, and the processor or processors may execute the program code and/or manipulate the program code, data, and/or algorithms. Accordingly, exemplary computer 14000 as shown in Fig. 14 includes system bus 14050 which operatively connects two processors 14051 and 14052, random access memory (RAM) 14053, read-only memory (ROM) 14055, input output (I/O) interfaces 14057 and 14058, storage interface 14059, and display interface 14061. Storage interface 14059 in turn connects to mass storage 14063. Each of I/O interfaces 14057 and 14058 may be an Ethernet, IEEE 1394, IEEE 802.11, or other interface such as is known in the art. Mass storage 14063 may be a hard drive, optical disk, or the like. Processors 14057 and 14058 may each be a commonly known processor such as an IBM or Motorola PowerPC, or an Intel Pentium.

Computer 14000 as shown in this example also includes an LCD display unit 14001, a keyboard 14002 and a mouse 14003. In alternate embodiments, keyboard 14002 and/or mouse 14003 might be replaced with a pen interface. Computer 4000 may additionally include or be attached to card readers, DVD drives, or floppy disk drives whereby media containing program code may be inserted for the purpose of loading the code onto the computer. In accordance with the present invention, computer 14000 may be programmed using a language such as Java, Objective C, C, C#, or C++ according to methods known in the art to perform those operations described above.

Accordingly, the above described user terminal could be, for example, a portable device comprising a StrongARM processor, an integrated touch-sensitive color screen with the ability to receive DVB-T broadcasts and, in some embodiments, the ability to send and receive GSM, PCS, or other cellular transmissions. The device could use an operating system such as Microsoft Windows CE or Symbian EPOC, perhaps with support for Java.

As noted above, the bandwidth of an incoming DVB-T datastream is approximately 22 Mbit/s. In certain embodiments a user might view content using a general purpose computer or other device that interfaces with a DVB-T receiver via a data connection whose bandwidth is less than 22 Mbit/s. This could be the case, for example, if a personal computer interfaced with a DVB-T receiver using a data connection such as a Universal Serial Bus (USB) interface, as USB offers a bandwidth on the order of 5 Mbit/s.

Under such circumstances, it may be desirable to have the DVB-T receiver partition incoming data into channels of narrow enough bandwidth to fit though the USB or other data connection. Different partitioning models are envisioned.

According to one embodiment, the receiver could break the incoming DVB-T datastream into channels of identical bandwidth, each possessing bandwidth equal to approximately the bandwidth of the data connection. As noted above, the bandwidth of a USB data connection is approximately 5Mbit/s. Therefore a 22 Mbit/s datastream could be broken down by the receiver into 5 channels of 4.4 Mbit/s each. Such partitioning would allow only one such channel to be transferred over the data connection at a time.

According to another embodiment, the receiver could break the incoming DVB-T datastream into channels of identical bandwidth, each possessing bandwidth equal to a small percentage of the bandwidth of the data connection. Thus, for example, when the data connection is a USB connection, a 22 Mbit/s DVB-T datastream could be broken down by the receiver into